INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research

International Journal of Sensor Networks

ISSN online: 1748-1287 - ISSN print: 1748-1279
https://www.inderscience.com/ijsnet

NASA space station rolling bearings anomaly detection based on PARA-LSTM model

Yingqian Zhang, Jiaye Wu, Hui Xie, Rongru Hua, Qiang Li

DOI: 10.1504/IJSNET.2023.10060575

Article History:

Received:	01 September 2023
Last revised:	03 September 2023
Accepted:	07 September 2023
Published online:	30 January 2024

Copyright © 2024 Inderscience Enterprises Ltd.

NASA space station rolling bearings anomaly detection based on PARA-LSTM model

Yingqian Zhang

School of Civil Engineering,
Sichuan University of Science and Engineering,
Zigong 643000, China
Email: zyq13568328215@163.com

Jiaye Wu*

School of Mechanical and Electrical Engineering,
Southwest Petroleum University,
Chengdu 610500, China
Email: wujy@scentralit.com
*Corresponding author

Hui Xie, Rongru Hua and Qiang Li

Technology Center,
Sichuan Shengtuo Testing Technology Co., Ltd.,
Chengdu 610045, China
Email: xieh@scentralit.com
Email: huarr@scentralit.com
Email: liqiang@scentralit.com

Abstract: Anomaly detection in time series data identifies abnormal events or behaviours. Traditional methods include principal component analysis (PCA) combined with Mahalanobis distance and long short-term memory (LSTM). Autoencoders and neural network techniques have been applied to the problem of anomaly detection. Still, challenges remain, such as large training data volume, network parameter initialisation, low training efficiency, and poor anomaly detection performance. This paper proposes an anomaly detection method based on parallel-long short-term memory (PARA-LSTM), which constructs two parallel processing structures. The method was tested on the rolling bearing vibration dataset collected by the NASA space station. It could detect anomalies five days ahead of the actual system destruction time, outperforming the PCA method by detecting anomalies one day earlier. PARA-LSTM has good performance, stability, and generalisation ability.

Keywords: autoencoder; bearing vibration; anomaly detection; Mahalanobis distance; autoencoder network; parallel-long short-term memory; PARA-LSTM.

Reference to this paper should be made as follows: Zhang, Y., Wu, J., Xie, H., Hua, R. and Li, Q. (2024) 'NASA space station rolling bearings anomaly detection based on PARA-LSTM model', *Int. J. Sensor Networks*, Vol. 44, No. 1, pp.49–61.

Biographical notes: Yingqian Zhang is an Associate Professor at the School of Civil Engineering, Sichuan University of Science and Engineering. He received his Master of Science and Bachelor of Science degrees from Sichuan University, China, in 2005 and 2002, respectively. His areas of interest are artificial intelligence, structural health monitoring, and non-destructive testing.

Jiaye Wu is a Professor at the School of Mechanical and Electrical Engineering at Southwest Petroleum University. He received his PhD from Tsinghua University, China in 1998. His areas of interest are geomechanics, non-destructive testing, and bridge health testing.

Hui Xie is currently a Deputy Chief Engineer of Sichuan Shengtuo Testing Technology Co., Ltd. He received his Bachelor's degree from Chengdu University of Technology, China 2013. His areas of interest are non-destructive testing and bridge health testing.

Rongru Hua is currently a Deputy Chief Engineer of Sichuan Shengtuo Testing Technology Co., Ltd. She received her Bachelor's degree from the Sichuan University of Science and Engineering, China, in 2010. Her areas of interest are non-destructive testing and business management.

Copyright © 2024 Inderscience Enterprises Ltd.

Qiang Li is currently a Deputy Chief Engineer of Sichuan Shengtuo Testing Technology Co., Ltd. He received her Bachelor's degree from the Sichuan University of Science and Engineering, China in 2020. His areas of interest are non-destructive testing and structural health monitoring.

1 Introduction

1.1 Introduction to anomaly detection

In recent years, computer and sensor technologies have undergone rapid advancements, with the concept of intelligence (e.g., Haug and Drazen, 2023) permeating various domains. Modern industrial systems are evolving towards larger and more complex structures. Data that reflects the operational status of equipment possesses the characteristics of big data (e.g., Shen et al., 2023), including volume, undiscovered patterns, abrupt changes, multiple modes, heterogeneity, and sparsity. Consequently, traditional anomaly detection algorithms are inadequate for meeting the anomaly detection requirements of industrial big data features in this new era.

Anomaly detection (e.g., Han et al., 2022; Ozdemir and Xiao, 2013; Sun et al., 2013) involves identifying data points that deviate significantly from most data points. Unlike problems with deterministic rules or obvious patterns, anomaly detection deals with rare events that are infrequent, difficult to predict, and uncertain. This gives rise to several unique complexities:

1 Uncertainty: anomalies are associated with numerous unknown factors, exhibiting unknown behaviours, data forms, and distributions. Hence, anomalies cannot be fully predicted until they occur.

2 Heterogeneity: anomalies lack deterministic rules or patterns. The features displayed by one anomaly are likely to be completely different from those exhibited by other anomalies.

3 Class imbalance (e.g., Santos et al., 2022): gathering and accurately labelling anomalous data points is exceedingly challenging. Ample normal data points are often available, while anomalous ones are scarce. The severe class imbalance problem makes it impractical to apply current supervised learning algorithms that perform well directly.

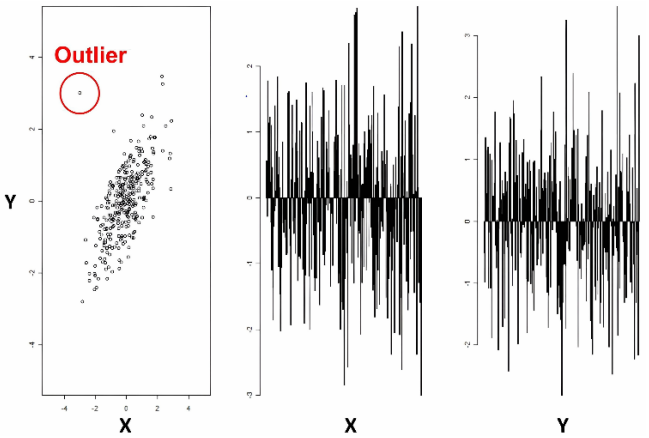
4 Low tolerance: misclassifying an anomalous data point as normal typically incurs much higher costs than misclassifying a normal data point as anomalous.

5 Various types of anomalies: currently, the field of anomaly detection explores three distinct types of anomalies:

- Point anomalies (e.g., Fisch et al., 2022): a few individuals differ from most others, such as health indicators in patient data.
- Conditional anomalies (e.g., Gudovskiy et al., 2022): data instances are anomalous under specific contextual conditions; otherwise, they are considered normal, such as a significant financial flow during a particular period.
- Collective anomalies (e.g., Shayegan et al., 2022): in a collection of data instances, an individual data point may not be anomalous, but a subset of data instances portrays anomalous behaviour, as seen in network fraud where some social accounts appear normal individually but form an anomalous group chat when combined.

In data analysis, anomaly detection (outlier detection) involves identifying points that significantly differ from most data and do not conform to a clear definition. Anomaly detection techniques find applications in various fields, including intrusion detection, fraud detection (e.g., Zhang et al., 2022), fault detection, system health monitoring, sensor network event detection, and ecosystem disruption detection. It is often employed to eliminate outlier data during the preprocessing stage. In supervised learning, datasets with removed outlier data often demonstrate statistically significant improvements in accuracy. Figure 1 provides an illustration of outlier detection for two variables.

Figure 1 Anomaly detection of two variables (see online version for colours)



When dealing with two-dimensional data (X and Y), identifying outliers visually becomes effortless by locating data points outside the typical scatter plot distribution. However, as depicted in the right-hand plot of Figure 1, directly identifying outliers by examining only one variable at a time is impossible. Outliers can be easily identified only when considering the combination of X and Y variables, as shown in the left-hand plot of Figure 1. As we expand from two variables to a significantly larger number, such as ten or even 100 times, the complexity of the problem intensifies.

This complexity is precisely encountered in real-world applications of anomaly detection.

1.2 Literature review on bearing anomaly detection

Bearings (e.g., Zhao et al., 2021), especially rolling bearings (Zhou et al., 2022), are essential components of various industrial machinery and equipment. They play a crucial role in ensuring the smooth operation of industrial processes and equipment safety. A bearing is a component that supports the rotational movement of mechanical bodies, reducing friction and ensuring rotational accuracy. Bearings can be classified based on the direction of load or nominal contact angle: radial bearings and thrust bearings. They can also be categorised according to the type of rolling elements: ball bearings and roller bearings. Furthermore, bearings can be classified based on their ability to accommodate misalignments: self-aligning and rigid bearings. Additionally, bearings can be classified based on the number of rolling elements: single-row, double-row, and multi-row bearings. They can also be divided into separable and non-separable bearings based on their component parts. Furthermore, bearings have various structural types, including those with or without filling grooves, with or without inner and outer rings, collar shapes, retaining edge structures, and the presence of cages. Lastly, bearings can be categorised based on their outer diameter size: miniature bearings (<26 mm), small bearings (28–55 mm), small and medium-sized bearings (60–115 mm), medium and large-sized bearings (120–190 mm), large bearings (200–430 mm), and extra-large bearings (>440 mm). Figure 2 depicts a physical diagram of a rolling bearing. However, when subjected to prolonged periods of high-intensity and high-load working conditions, bearings are prone to experiencing abnormal conditions, leading to failures. Accurately and timely detecting abnormal data can assist in promptly maintaining equipment and preventing serious accidents. In the era of big data, with the rapid advancements in sensor and machine learning technologies, there is a growing interest in utilising advanced theories and methods to extract features from historical state monitoring data. The construction of data-driven models for bearing abnormality detection aims to ensure the accuracy and stability of such detection. This field of research holds substantial academic value and practical significance.

Monitoring (e.g., Lu et al., 2021; Sun et al., 2006) the vibration data of bearings provides the most straightforward and convenient means of determining their abnormality. By utilising acceleration sensors, data can be collected easily and rapidly, allowing for real-time monitoring. Typically, vibration data is gathered by placing multiple sensors at various locations. Consequently, the characteristics of vibration data primarily encompass a large volume of data, low dimensionality, and a certain level of correlation between dimensions.

Figure 2 Physical drawing of a typical rolling bearing (see online version for colours)

Various outlier detection methods (e.g., Nayak and Perros, 2020; Wu et al., 2015; Ding and Feng, 2021) detect bearing vibration abnormalities. These methods primarily include:

1 Statistical methods: this method assumes that most normal data follows a specific distribution, while abnormal data deviate from this distribution. However, determining the probability distribution model and its parameters can be challenging for this method.

2 Neighbour-based outlier detection methods: these methods, such as the K-nearest neighbour algorithm (KNN) (e.g., Uddin et al., 2022) and the local outlier factor algorithm (LOF) (e.g., Aubert et al., 2022), are straightforward to implement. However, they may perform poorly when bearing vibration data with limited abnormal samples and unbalanced datasets. The LOF algorithm addresses the performance issues of the KNN algorithm for unbalanced datasets by introducing local reachable density. Nonetheless, the LOF algorithm is highly sensitive to parameter selection and entails high algorithm complexity and time cost.

3 Support vector machine (SVM) (e.g., Tanveer et al., 2022): this method determines abnormal data by training the boundary or features of data objects. It necessitates a sufficient amount of normal data for training and requires appropriate parameter configuration, as the choice of parameters greatly impacts the detection outcome.

4 Isolation forest (iForest) (e.g., Tokovarov and Karczmarek, 2022) algorithm: this algorithm detects abnormal data by constructing isolation trees and exhibits linear time complexity. However, it adopts an unsupervised approach, meaning that it does not utilise labelled data for model training. Consequently, this can lead to lower detection accuracy, particularly when the data is unevenly distributed.

In recent years, deep neural networks (DNNs) (e.g., Liu et al., 2022) have been effectively utilised for automatic feature extraction and recognition. DNNs can adaptively extract valuable and significant features, simplifying the complex and challenging feature extraction process and exhibiting good generalisation. Autoencoders (AEs) were initially introduced by Rumelhart (1993) for processing

intricate data. Subsequently, Hinton et al. proposed deep-learning neural networks, leading to the development of deep autoencoders. Building upon deep autoencoders, Ng proposed high-dimensional and sparse hidden layers, incorporating sparsity constraints to enhance the feature learning capability of autoencoders, thus introducing sparse autoencoders. Autoencoders and deep neural networks are widely applied in fault diagnosis, image recognition, and anomaly detection. These techniques have also been extensively researched and applied to address the anomaly detection issue, including detecting anomalies in industrial data such as bearing vibration. Autoencoders have garnered much attention and utilisation due to their exceptional feature extraction capabilities.

Advancements in sensor technology (e.g., Sun et al., 2007a, 2007b) have facilitated the easier and quicker collection of bearing vibration data. However, this has resulted in a substantial increase in data volume. Handling large amounts of data can lead to reduced training efficiency and challenges in enhancing detection accuracy during the training process of autoencoders and neural networks. Extensive research on the LSTM (e.g., Kumar et al., 2022) algorithm has revealed that its modelling capability for periodic data surpasses its modelling capability for irregular data. Bearing sensor data falls under the category of irregular data, which differs significantly from LSTM-based predictions for periodic time series. Through extensive research and experimentation, we have developed a deeper comprehension of the intricacies of LSTM. We are working towards improving its network structure to enhance its ability and accuracy for bearing sensor time-series predictions.

The proposed enhancement involves utilising two parallel network structures during training to prevent overfitting the abnormal dataset. Liu et al. connected two LSTM networks in a cascading manner, where the prediction value from the first network and the residual of the actual value were utilised as inputs for the second network. The objective was for the second network to learn the remaining information and patterns within the residual value. Building upon this idea, we propose the improved PARA-LSTM network and construct the network model, albeit in a parallel manner instead of a cascading one. Specifically, this paper establishes a network with two parallel processing structures: a 'health model' network and a 'state model' network. The parallel network structure effectively addresses the issue of LSTM's detection capability diminishing in certain data due to its inability to learn the 'normal state' of the device.

The major contributions of this paper are summarised as follows:

- A parallel long short-term memory (PARA-LSTM) based anomaly detection method is proposed. The proposed method constructs two parallel processing networks: a 'health model' network and a 'state model' network. The outputs of these two parallel networks are then fed into a fully connected layer using ensemble voting to generate anomalous data. Finally, the predicted outputs are produced.
- Three anomaly detection methods are applied to the bearing vibration dataset collected by the Intelligent maintenance system (IMS) provided by NASA. The methods include principal component analysis (PCA) with Mahalanobis distance, traditional LSTM, and the proposed PARA-LSTM-based anomaly detection method. Experimental results show that the PARA-LSTM method can detect anomalies on 2004-02-14, one day earlier than the PCA method using the Mahalanobis distance model. Therefore, the proposed PARA-LSTM method outperforms other anomaly detection methods, exhibiting stability and generalisation capabilities.
- The anomaly detection analysis of the bearing vibration dataset collected by the IMS provided by NASA demonstrates the superiority of the proposed PARA-LSTM method over conventional methods. Furthermore, the method can be applied in other time series-related fields, such as medical electrocardiograms and stock price prediction.

The rest of the paper is organised as follows: in Section 2, theoretical foundations needed for anomaly detection, including PCA, Mahalanobis distance, and long short-term memory (LSTM), are introduced. Section 3 presents the construction of the PARA-LSTM model. Section 4 applies the three anomaly detection methods to NASA's rolling bearing data. Finally, Section 5 summarises the research findings and presents an outlook.

2 Theoretical background

A complex device's health cannot be assessed solely on a single measurement. We must consider a combination of different measurement methods to develop a true understanding of its condition.

2.1 PCA for dimensionality reduction

PCA (e.g., Elhaik, 2022) is a common dimensionality reduction technique for high-dimensional data. It extracts principal components – directions with maximum data variance. PCA performs a linear transformation projecting correlated variables onto a smaller orthogonal set that retains most of the original variance. PCA is one of the most widely used dimensionality reduction approaches, as it can handle high-dimensional, noisy, and correlated data.

Training samples must consist of positive samples with zero mean. We apply z-score normalisation. Let the training samples be X_{n*m}, where n is the number of samples and m is the number of features. The covariance matrix of the m features is calculated as shown in equation (1):

$$\sum_{m\times m} \frac{1}{n-1} X^T X \tag{1}$$

We then calculate the eigenvalues λ_i and eigenvectors p_i of the covariance matrix, arranging the eigenvalues in descending order as shown in equation (2):

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m \tag{2}$$

We reorder the eigenvectors according to their corresponding eigenvalues, as shown in equation (3):

$$V_{m*m} = [p_1, p_2, \ldots, p_m] \tag{3}$$

The first k eigenvalues are chosen for PCA dimensionality reduction according to the selected principle. Then, the first k eigenvalues are arranged in a diagonal matrix S_{k*k} as shown in equation (4), and the k corresponding eigenvectors form the dimensionality reduction matrix P_{m*k} as shown in equation (5):

$$S_{k*k} = diag(\lambda_1, \lambda_2, \ldots, \lambda_k) \tag{4}$$

$$P_{m*k} = [p_1, p_2, \ldots, p_k] \tag{5}$$

After dimensionality reduction, while the number of samples stays the same at n, the number of features is reduced to k. The dimensionality reduction formula is shown in equations (6)~(7):

$$\tilde{X} = XP \tag{6}$$

$$X' = \tilde{X}PT = XPPT \tag{7}$$

X' is the matrix obtained by dimensionality reduction of X.

The steps for performing fault detection using the PCA method are as follows:

1 Build a normal principal component model
   - Step1 Standardise the normal data, turning it into a dataset with a mean of 0 and a variance of 1.
   - Step 2 Use the dataset in Step 1 as the training dataset to build a PCA principal component model and extract principal components.
   - Step 3 Calculate the training dataset's PCA statistics and control limits.

2 Fault or anomaly detection and diagnosis
   - Step 1 Obtain the test dataset and standardise it.
   - Step 2 Calculate the T^2 statistic for the standardised data and compare it with the control limits for the normal state. If it exceeds the control limits, it is considered abnormal. Otherwise, it is considered normal.

2.2 Mahalanobis distance

Renowned Indian statistician P.C. Mahalanobis (e.g., Colombo et al., 2022) introduced a generalised distance measure known as Mahalanobis distance (MD), which accounts for the correlation between variables. The primary concept involves utilising the covariance matrix between vectors to characterise their MD. MD is a widely employed distance metric in machine learning used to assess the similarity between data points, much like Euclidean distance, Manhattan distance, and Hamming distance. However, it specifically addresses the issue of non-independent and non-identically distributed dimensions within high-dimensional linear datasets.

For a dataset $X = (X_1, X_2, \ldots, X_n)$ with n data points with m dimensions, a mean $\mu = (\mu_1, \mu_2, \ldots, \mu_m)^T$, and covariance matrix Σ, and one data point $x = (x_1, x_2, \ldots, x_m)^T$, the Mahalanobis distance is given by as shown in equation (8):

$$D_M(x) = \sqrt{(x-\mu)^T \Sigma^{-1} (x-\mu)} \tag{8}$$

The Mahalanobis distance can be interpreted as the distance between a data point and the mean of the population data, where Σ^{-1} is the inverse of the covariance matrix Σ.

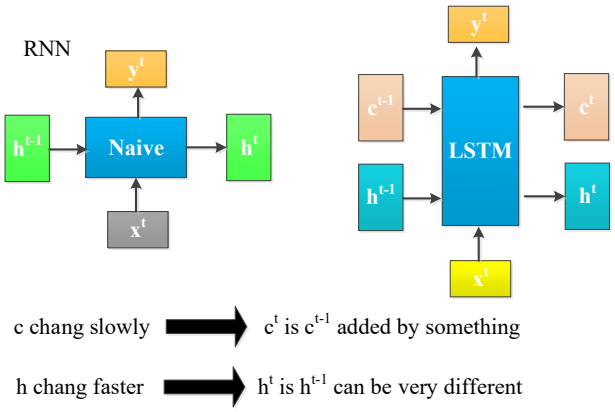
2.3 LSTM

Recurrent neural networks (RNNs) are a type of neural network utilised for sequential data processing. Compared to standard neural networks, RNNs can effectively handle data exhibiting temporal changes.

LSTM is a specialised variant of RNNs specifically designed to address the challenge of gradient vanishing and exploding during extensive sequence training. In simpler terms, LSTMs outperform regular RNNs when operating on lengthy sequences.

Figure 3 depicts the primary input and output distinctions between the LSTM structure (located on the right-hand side of the illustration) and a typical RNN.

Figure 3 LSTM structure vs. regular RNN structure (see online version for colours)



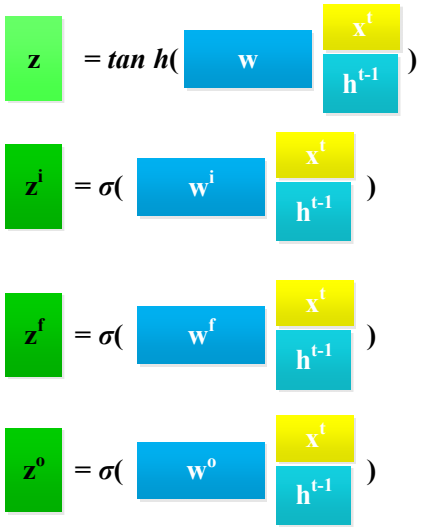
In contrast to RNN, which only has a single state to propagate, LSTM incorporates two states for propagation: the cell state (c^t) and the hidden state (h^t).

LSTM leverages the current input (x^t) and the previous propagated state (h^{t-1}) to train and acquire four distinct states, as illustrated in Figure 4.

Among these, z^f, z^i, and z^o represent gate states produced by multiplicatively combining concatenated vectors with weight matrices, subsequently transformed into values ranging from 0 to 1 using a sigmoid activation function. On the other hand, z is transformed into values ranging from –1 to 1 using a hyperbolic tangent (tanh) activation function

(tanh is utilised since it is considered input data rather than gate signals).

Figure 4 LSTM structure with four states (see online version for colours)



3 Methodology

3.1 Method 1: PCA + Mahalanobis distance

The Mahalanobis distance is a distance metric that considers the covariance matrix of the dataset. It is a more robust distance measure than the Euclidean distance because it considers the correlation between data features. In a dataset, a data point with a small Mahalanobis distance is more similar to the dataset mean than a data point with a large Mahalanobis distance. Features are often correlated in low-dimensional industrial datasets like bearing vibration data due to the manufacturing process and data acquisition equipment. Therefore, the Mahalanobis distance is a more suitable distance metric for bearing vibration data.

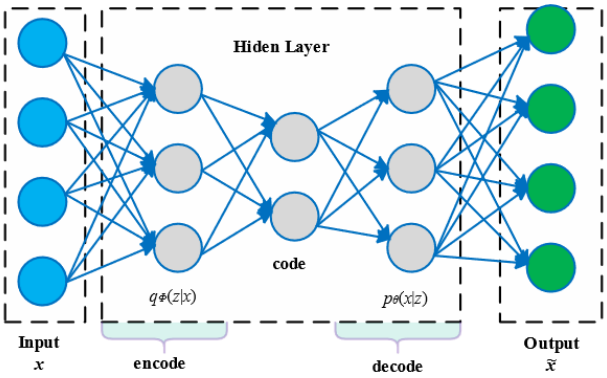
To classify a test point into one of N categories using the Mahalanobis distance, we first estimate the covariance matrix of each category based on known samples that belong to that category. Since we are only interested in 'normal' versus 'abnormal' classification in this paper, we use training data containing only normal operating conditions to calculate the covariance matrix. Then, for a given test sample, we calculate the Mahalanobis distance of the sample to the 'normal' class. The test point is classified as 'abnormal' if the distance exceeds a certain threshold.

3.2 Method 2: autoencoder

An autoencoder is an artificial neural network that learns efficient data encoding unsupervised. Its goal is to learn an 'encoding' of a set of data – typically used for dimensionality reduction. Both the encoder and decoder learn so the autoencoder tries to generate an encoding as similar as possible to its original input from the reduced-dimensional representation.

Structurally, the simplest form of an autoencoder is a feedforward neural network, unlike a recurrent neural network. It resembles a multilayer perceptron (MLP) with an input layer, an output layer, and one or more hidden layers connecting them. However, the autoencoder's output layer has the same number of nodes as the input layer, and its goal is to reconstruct its input, as shown in Figure 5.

Figure 5 Autoencoder network (see online version for colours)



The basic idea of anomaly detection and condition monitoring is to use autoencoders to compress sensor readings into a lower-dimensional representation that captures the correlations and interactions between various variables.

3.3 Method 3: anomaly detection method based on PARA-LSTM

PARA-LSTM is a neural network model that consists of two networks: a health model network and a state model network. The health model is a two-layer LSTM network that inputs all LSTM layers' output states (h1, h2, …, ht) as input and outputs them to a fully connected layer. The fully connected layer outputs a single unit, which is the prediction of the network. The network's input is now only the X vector rather than the X and Y vectors of the previous section. This network can be considered as an encoder-decoder that reconstructs normal data.

We feed the normal X vectors as input to the network and expect it to predict the next value of Y. By feeding the normal data X to train the model, the model should be able to reconstruct 'normal' data Y. In other words, under normal X data, the model should learn to output the value of the next Y that it believes is typical.

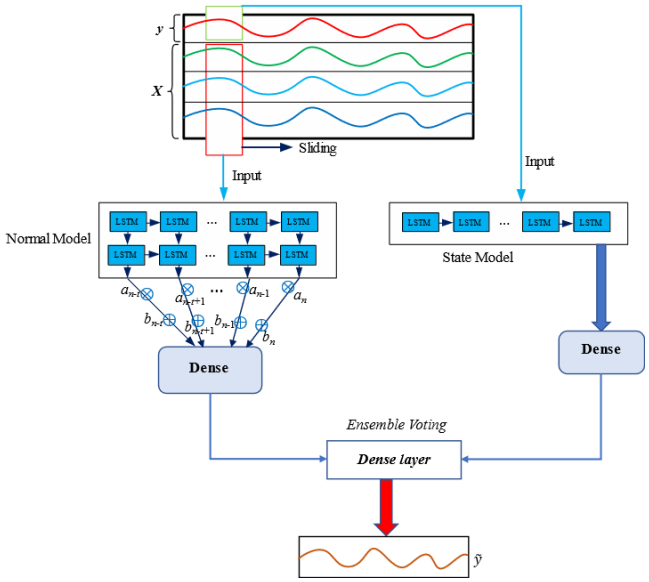
However, if Y vectors are not introduced for training, the model cannot learn how historical Y vector values impact the trend of predicted values over time. When the health model is trained alone and used for prediction, it will produce globally biased predictions for some datasets. Therefore, we add a parallel-trained state model.

The state model is a single-layer LSTM network that only uses the last output time ht as output. The input vector is the Y vector, and the output is the predicted next Y value. This network can be considered a state-preserving model that learns the historical features of Y vectors. After training, the state model should be able to correctly judge

the next Y's development trend based on Y's historical values.

Finally, the outputs of the two parallel networks are input to another fully connected layer in an ensemble voting manner to generate the final predicted output Y. The model diagram is shown in Figure 6.

Figure 6 PARA-LSTM model diagram (see online version for colours)



LSTM is a recurrent neural network (RNN) model with space and time complexity of O(n), where n represents the sequence length. The PARA-LSTM model proposed in this paper consists of two parallel LSTM models, resulting in a space complexity of O(2n).

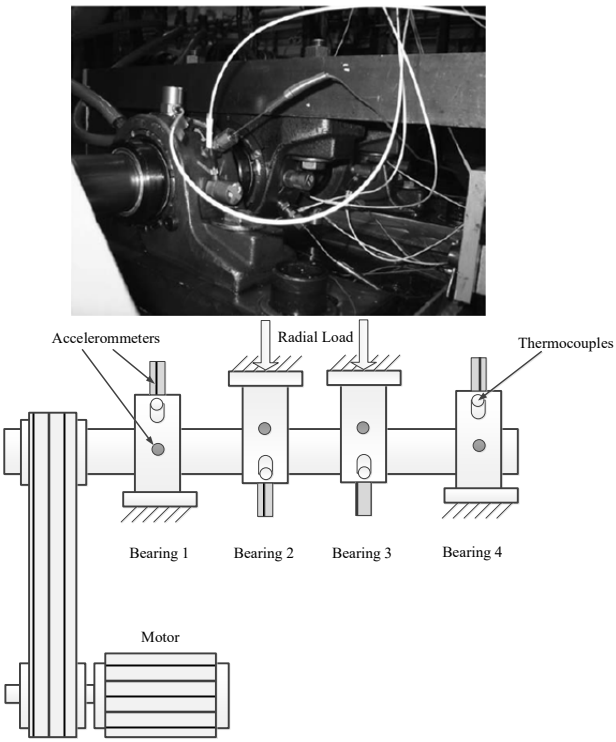
4 Validation and analysis

Three methods, namely PCA combined with Mahalanobis distance, an LSTM model, and a PARA-LSTM model, were employed to conduct anomaly detection analysis on the bearing vibration dataset obtained from the intelligent maintenance system provided by the National Aeronautics and Space Administration (NASA). The system is depicted in Figure 7.

Table 1 Bearing data collection information

Date	*Bearing 1 (N.m/s)*	*Bearing 2 (N.m/s)*	*Bearing 3 (N.m/s)*	*Bearing 4 (N.m/s)*
2004/2/12 10:32	0.0583	0.0718	0.0832	0.0431
2004/2/12 10:42	0.0590	0.0740	0.0844	0.0445
2004/2/12 10:52	0.0602	0.0742	0.0839	0.0444
2004/2/12 11:02	0.0615	0.0738	0.0845	0.0451
2004/2/12 11:12	0.0614	0.0756	0.0828	0.0451
…	…	…	…	…
2004/2/19 6:22	0.0012	0.0008	0.0007	0.0017

Figure 7 Bearing four sensor diagram



4.1 Data preparation

For this study, we will utilise vibration sensor readings from the NASA Acoustics and Vibration Database as our dataset. The NASA study involved collecting sensor readings from four bearings operated under a constant load until failure occurred over multiple days. Our dataset comprises individual files containing 1-second snapshots of vibration signals recorded at 10-minute intervals. Each file contains 20,480 sensor data points for each orientation, obtained from orientation sensors with a sampling rate of 20,000 Hz. Table 1 illustrates that data collection for the bearings spanned from 10:32 AM on 12 February 2004, to 6:22 AM on 19 February 2004, totalling nearly seven days. Figure 8 depicts the time-varying curves of the bearing vibration data. As seen in Figure 8, bearing 1 displayed a notable abnormal fluctuation after 17 February, while the other three bearings showed more pronounced fluctuations after 18 February until the entire system eventually failed on 19 February.

For time series data, the division between the training and test sets is primarily based on a specific time point. In this study, we employed the data from an initial normal period, specifically 222 data points recorded between 11:02 AM on 12 February 2004, and 1:23 PM on 13 February 2004, as the training data. Subsequently, we utilised all the subsequent data, comprising 760 data points from 1:23 PM on 13 February 2004, to 6:22 AM on 19 February 2004, as the test data. Figure 9 illustrates the training dataset, whereas Figure 9 represents the test dataset.

Figure 8 Time-varying trend of bearing sensor data (see online version for colours)

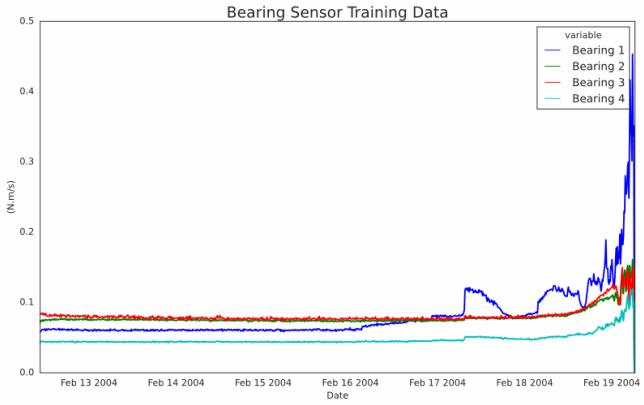
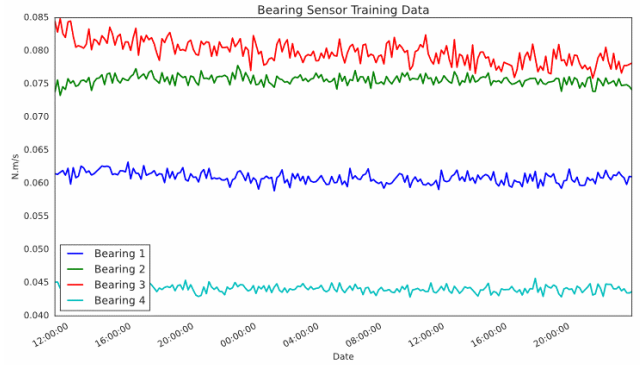


Figure 9 Bearing sensor data training set (see online version for colours)



As depicted in Figure 9, during the initial stage, which corresponds to the training data phase, the vibration data from all four bearings exhibit fluctuations yet remain within a reasonable range. In Figure 10, representing the later stage or test data phase, the vibration data of the four bearings show distinct abnormalities before the occurrence of destructive failure. Bearing 1, in particular, displays conspicuous abnormalities at an earlier stage, with significant abnormal fluctuations noted after 17 February.

Figure 10 Bearing sensor data test set (see online version for colours)

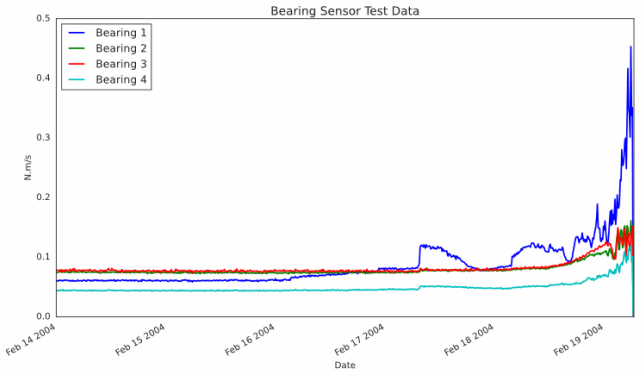


Figure 11 Schematic diagram of Fourier transform from time domain to frequency domain (see online version for colours)

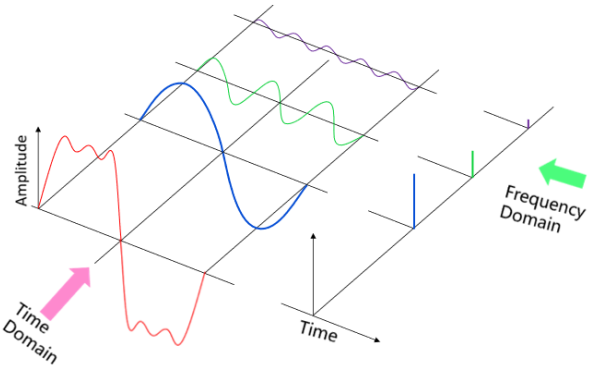


Figure 12 Bearing sensor frequency domain training set (see online version for colours)

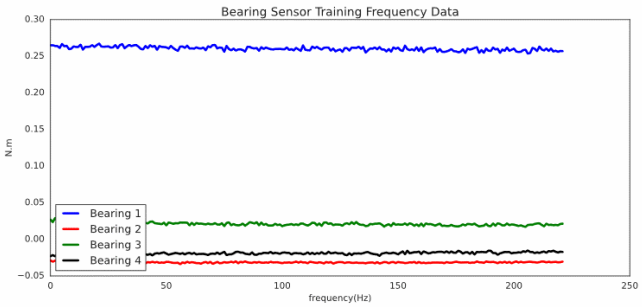
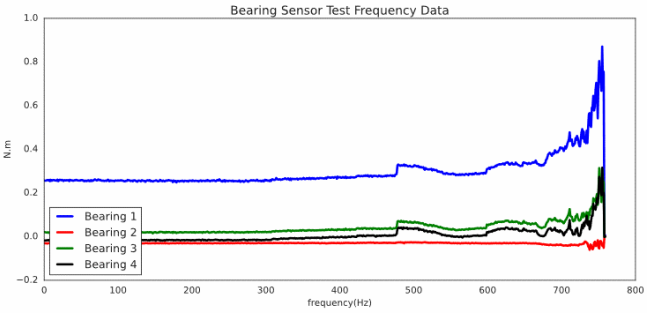


Figure 13 Bearing sensor frequency domain test set (see online version for colours)



Within the test set timeframe, there is a notable shift in the sensor pattern. As the failure point approaches, the vibration readings in the bearings intensify and exhibit violent oscillations. We will utilise the Fourier transform to convert the signal from the time domain to the frequency domain to gain a slightly different perspective on the data. Figure 11 visualises applying the Fourier transform to convert the signal. Figures 12 and 13 illustrate the frequency domain curves of the bearing sensor training set and test set, respectively. Figure 12 shows that during the training phase, the bearing sensor performance remains smooth in the frequency domain, just like in the time domain. Figure 13 reveals that bearing one and bearing 3 exhibit distinct abnormalities near 480 Hz, while bearing four shows abnormalities near 720 Hz, and bearing three exhibits abnormalities near 750 Hz. As the failure time approaches, the frequency amplitude and energy of the four bearings experience a more pronounced increase.

4.2 PCA and MD

The bearing sensor data is relatively clean and does not require any cleaning. Therefore, the data preprocessing involves a simple normalisation process. The normalisation is performed using the min-max normalisation method. The resulting normalised data is presented in Table 2.

PCA extracts the principal components from the bearing sensor data. Initially, the number of principal components is set to 2. In this dataset, it is possible to increase the number of principal components to three for satisfactory results. However, given that the original dataset consists of four components, specifically the vibration data from four bearings, setting the number of principal components to three does not effectively accomplish the goal of dimensionality reduction. Therefore, this study sets the number of principal components to two, effectively reducing the original four-dimensional data to a two-dimensional representation. Table 3 presents the statistical results of the principal component analysis.

As shown in Table 3, principal component 1 accounts for 51.0% of the information, while principal component 2 only accounts for 20.4%. Together, principal component 1 and principal component 2 represent a total of 71.4% of the data information. It is worth noting that projecting the four-dimensional data to two dimensions results in a loss of 28.6% of the information.

Table 2 Normalised bearing data

Date	*Bearing1*	*Bearing 2*	*Bearing 3*	*Bearing 4*
2004/2/12 11:02	0.5946	0.1250	0.9532	0.7947
2004/2/12 11:12	0.5728	0.5152	0.7733	0.8078
2004/2/12 11:22	0.6430	0.0000	1.0000	0.4723
2004/2/12 11:32	0.7074	0.2906	0.7500	0.6447
2004/2/12 11:42	0.5427	0.1974	0.6829	0.3545
…	…	…	…	…
2004/2/14 00:02	0.3546	0.2278	0.2445	0.4254
2004/2/14 00:12	0.2497	0.3099	0.2311	0.2544
2004/2/14 00:32	0.6108	0.5028	0.1623	0.3804
…	…	…	…	…

Figures 14 and 15 depict the trend of principal components in the bearing sensor training set and test set, respectively. In Figure 13, principal components 1 and 2 exhibit relative stability during the training stage, with no noticeable abnormal fluctuations observed. Conversely, Figure 15 reveals that principal component 1 starts displaying abnormalities on 16 February, which becomes more pronounced on 17 February. Principal component 2, on the other hand, does not exhibit any significant abnormalities until around 19 February.

Table 3 demonstrates that principal component 1 encompasses 51.0% of the information in the bearing sensor data, indicating its higher information content compared to principal component 2. This aligns with the findings from Figure 15, which indicate that the principal component with greater information content can detect faults at an earlier stage.

Figure 14 Principal components trend plot of sensor training data (see online version for colours)

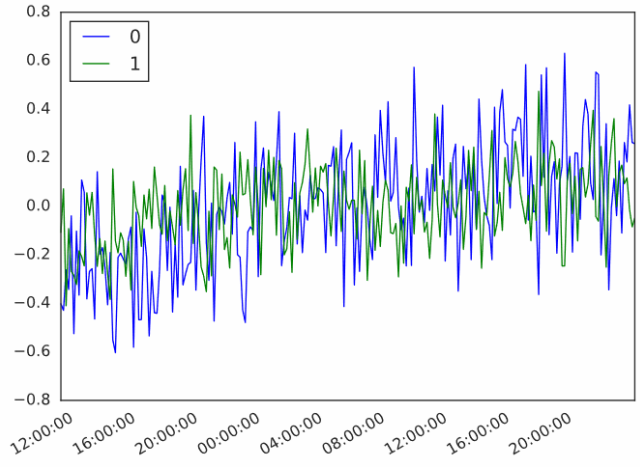


Figure 15 Principal components trend plot of sensor test data (see online version for colours)

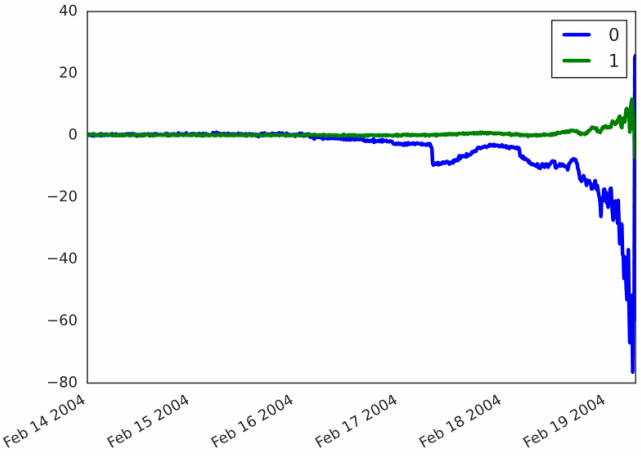


Table 3 Statistics on the proportion of each principal component

Principal component 1 (%)	*Principal component 2 (%)*	*Total (%)*
51.0	20.4	71.4

The calculation of the Mahalanobis distance involves three main steps: verifying if the matrix is positive definite, solving the covariance and inverse covariance matrices, and computing the Mahalanobis distance. To automatically determine the threshold of the Mahalanobis distance during normal operation, there are functions available for automated threshold calculation. However, the threshold itself needs to be appropriately adjusted based on the specific project requirements, and the function cannot automatically set it. The specific calculation process of the Mahalanobis distance includes: firstly, computing the covariance matrix and its inverse matrix based on the data in the training set; secondly, calculating the average value

of the input variables in the training set; and finally, utilising the average value to individually compute the Mahalanobis distance between the training set and the test set to the centre of gravity. If the input variables follow a normal distribution, the square of the Mahalanobis distance between the dataset and the distribution centre should follow a χ^2 distribution. This hypothesis forms the basis for determining the 'threshold' to identify outliers. However, as the data may not always conform to this hypothesis in certain cases, it is necessary to visualise the distribution of the Mahalanobis distance. Figures 16 and 17 present the square distribution of the Mahalanobis distance in the training data and the distribution of the Mahalanobis distance.

Figure 16 Square distribution of Mahalanobis distance in the training dataset (see online version for colours)

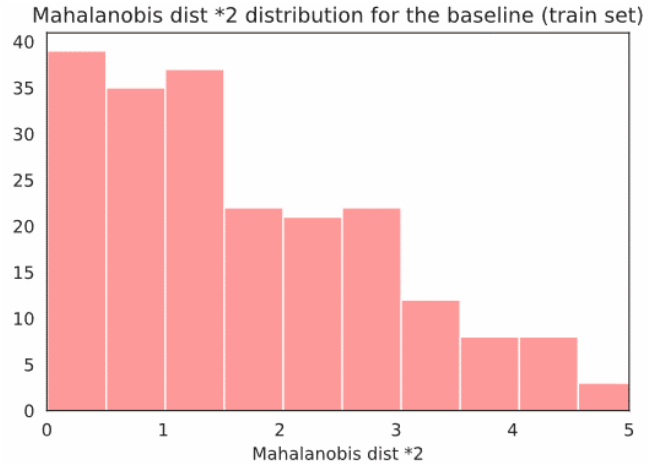
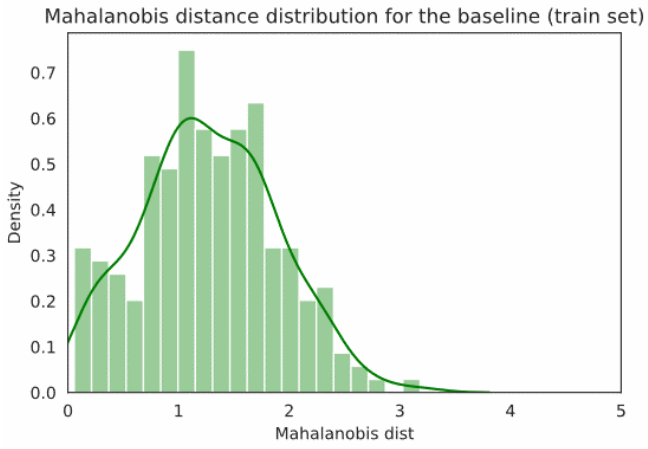


Figure 17 Distribution of Mahalanobis distance in training dataset (see online version for colours)



Based on the distribution of the Mahalanobis distance illustrated in Figure 17, the outlier threshold can be set to 4 standard deviations of the average Mahalanobis distance in the training data. Specifically, we select 3.812 as the critical value for the Mahalanobis distance. Points that exceed this critical value are classified as outliers, while points below the critical value are classified as normal. The training set comprises 222 normal points, and the test set consists of 760 points, of which 409 points are identified as outliers. This analysis demonstrates the adequacy of our data partition between the training and test sets.

Figure 18 Anomaly detection using Mahalanobis distance (see online version for colours)

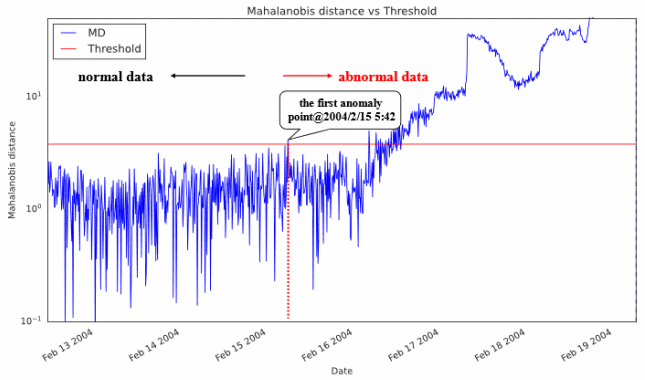


Figure 18 depicts the application of the Mahalanobis distance for bearing fault detection. In Figure 17, the initial fault was detected at 5:42 AM on 15 February, and the system failed at 6:22 AM on 19 February. This demonstrates that the PCA and MD method can identify faults four days before the vibration system experiences a failure. Consequently, this enables timely repairs and maintenance of the bearing monitoring system before failure. Furthermore, Figure 18 reveals that not all monitoring points following the initial fault detection at 5:42 AM on 15 February were considered abnormal. At 1:12 PM on 16 February, all monitoring data were unanimously classified as abnormal. Table 4 presents the first five instances when the system exhibited abnormal behaviour. The table reflects that although the system demonstrated abnormality during these five-time points, the severity of the abnormalities remained relatively low. This observation underscores the ability of the MD and PCA model to detect abnormalities before system failure.

Table 4 The first five time points when the MD and PCA model detected abnormalities

Date	*MD*	*Threshold*	*Anomaly*
2004/2/15 05:42:39	4.178	3.812	True
2004/2/16 04:12:39	5.031	3.812	True
2004/2/16 06:12:39	4.835	3.812	True
2004/2/16 06:52:39	4.101	3.812	True
2004/2/16 07:22:39	4.241	3.812	True

4.3 LSTM model

Using LSTM for time series anomaly detection (TSAD) involves two steps. The first step entails employing LSTM for time series prediction, while the second step entails utilising the discrepancy between the predicted and actual outcomes to determine the range of anomalies. Figure 19 displays the loss curve of the training set data. The horizontal axis (epoch) represents the number of training iterations, while the vertical axis denotes the training loss

value, measured by mean absolute error (MAE). Examining Figure 19, two notable features of the curve become apparent: a rapid decline and subsequent convergence. This behaviour arises due to the relatively small size of the training data, enabling the model to converge expeditiously. Figure 20 illustrates the distribution diagram of the loss function. Observing Figure 20, one can discern that the overall distribution of the loss function approximates a normal distribution. To establish the anomaly judgment threshold criterion, the 95% confidence level is selected, corresponding to a value of 0.285.

Figure 19 Loss iteration curve of the training set with the LSTM model (see online version for colours)

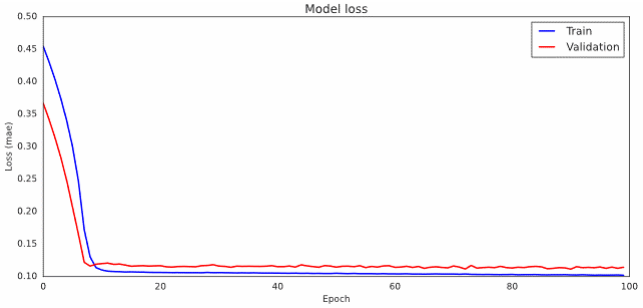


Figure 20 Loss distribution diagram with LSTM model (see online version for colours)

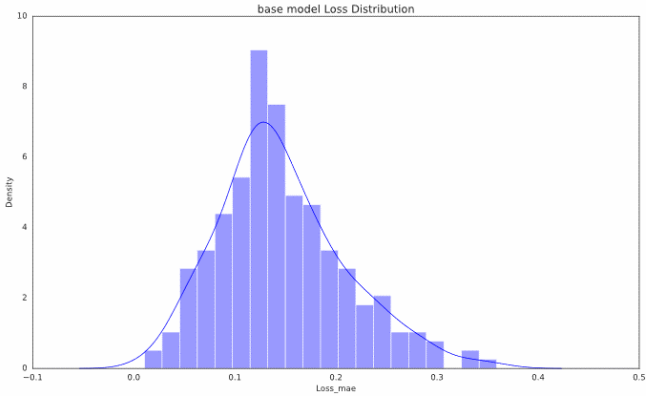


Figure 21 LSTM model uses threshold loss for anomaly detection (see online version for colours)

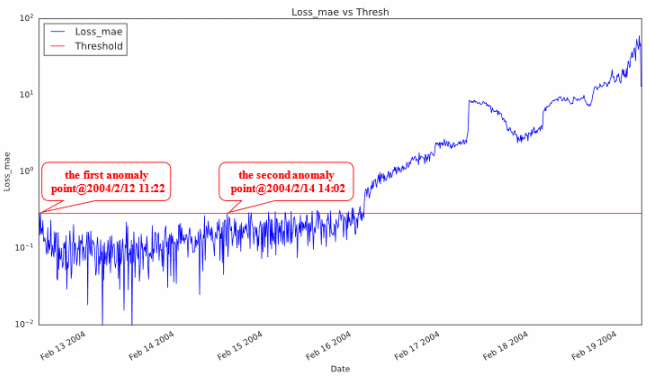


Table 5 The first five time points when the LSTM model detected anomalies

Date	*MD*	*Threshold*	*Anomaly*
2004/2/12 11:22	0.293	0.285	TRUE
2004/2/14 14:02	0.288	0.285	TRUE
2004/2/14 19:52	0.293	0.285	TRUE
2004/2/14 20:32	0.293	0.285	TRUE
2004/2/15 1:12	0.296	0.285	TRUE

Figure 21 depicts the application of the loss function threshold for anomaly detection. Observing Figure 21, it becomes evident that the LSTM model can detect anomalies in the system as early as 2:02 PM on 14 February, before system failure. However, the model also exhibited a clear misjudgement by indicating an anomaly at 11:22 AM on 12 February, directly contradicting the actual events. Therefore, the traditional LSTM model can detect anomalies before system failure, but it also displays instances of clear misjudgement. Table 5 enumerates the first five instances when the system exhibited abnormal behaviour. Notably, each time point listed in Table 5 occurs one day earlier than in Table 4. It is important to exclude the first erroneous anomaly point. This observation underscores the capability of the traditional LSTM model to identify anomalies before system failure.

4.4 PARA-LSTM

The proposed improved anomaly detection model, PARA-LSTM, in this paper, combines the outputs of two parallel networks using ensemble voting. These combined outputs are fed into another fully connected layer to generate the final prediction output. The model incorporates the ELU (exponential linear unit) activation function, as illustrated in Figure 22. For values of x greater than 0, the model returns x; for x less than 0, the model returns $\exp((x) - 1)$. Compared to RELU, ELU provides a non-zero output for $x < 0$, which promotes faster convergence of the model weights.

Figure 22 ELU activation function (see online version for colours)

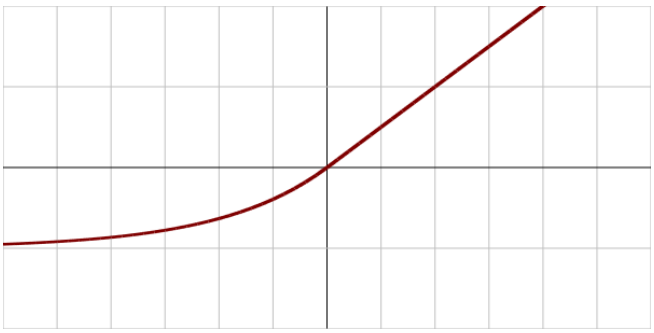


Figure 23 Loss iteration curve of the training set with the PARA-LSTM model (see online version for colours)

Figure 24 Loss distribution diagram with PARA-LSTM model (see online version for colours)

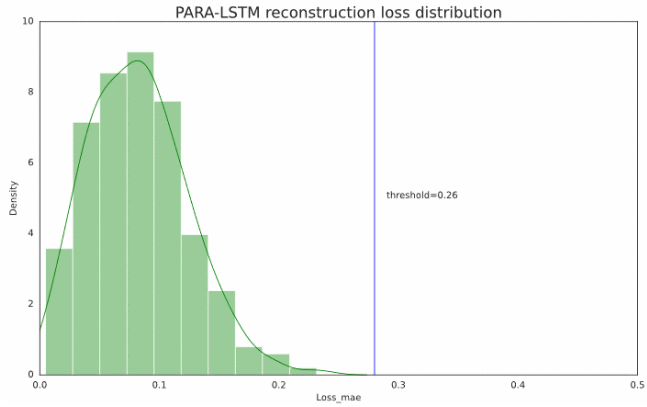


Figure 25 PARA-LSTM model uses threshold loss for anomaly detection (see online version for colours)

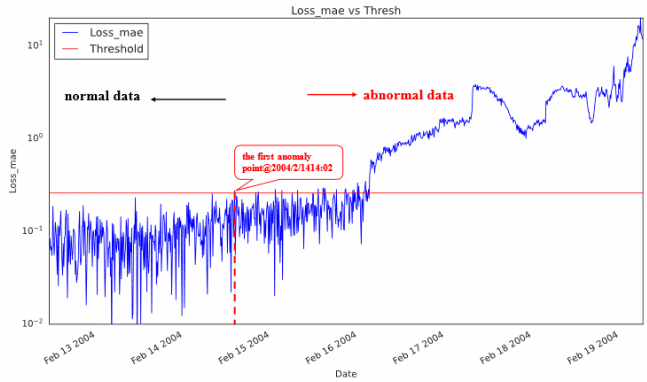


Figure 23 presents the curve of the loss function for the PARA-LSTM model during the iteration process. The figure illustrates a gradual decrease in the loss function, which tends to stabilise as the number of iterations increases, both in the training and validation sets. Moreover, Figure 23 reveals that 100 iterations are viable for the PARA-LSTM model. This allows for model convergence while effectively minimising the number of iterations required.

By visualising the distribution of the loss function in the training set, we can determine a suitable threshold for anomaly identification. Figure 24 displays the curve representing the distribution of the loss function, indicating that 0.26 is a more suitable threshold for identifying anomalies. Once the appropriate threshold has been selected, the loss function of the test set can be computed, enabling us to identify the time at which the anomaly occurred.

Table 6 The first five time points when the PARA-LSTM model detected anomalies

Date	*MD*	*Threshold*	*Anomaly*
2004/2/14 14:02	0.275	0.260	TRUE
2004/2/14 19:52	0.262	0.260	TRUE
2004/2/14 20:32	0.268	0.260	TRUE
2004/2/15 1:12	0.286	0.260	TRUE
2004/2/15 3:02	0.278	0.260	TRUE

Similar to the PCA method using the Mahalanobis distance, Figure 25 illustrates that the PARA-LSTM model effectively detects the system anomaly at 2:02 PM on 14 February. This detection is consistent with the second anomalous point identified by the traditional LSTM model, showcasing the absence of similar LSTM misjudgments in the PARA-LSTM model. Additionally, Table 6 presents the first five time points when the system experiences abnormalities. A comparison reveals that 4 of these five abnormal time points align perfectly with Table 5. This further demonstrates that the PARA-LSTM model successfully inherits the robust anomaly detection capability of the traditional LSTM model. The PARA-LSTM model proposed in this paper accurately detects anomalies nearly five days before system failure while avoiding potential misjudgements that could occur in the traditional LSTM model.

5 Conclusions

This paper explores the application of three models for analysing the issue of anomaly detection in NASA-bearing sensors. All three models yield similar results, successfully detecting anomalies before the bearing failure on 19 February. The MD and PCA model detected the earliest anomaly at 5:42 AM on 15 February, while the traditional LSTM and PARA-LSTM models identified the first anomaly at 2:02 PM on 14 February. Nonetheless, it is worth noting that the traditional LSTM model exhibited some misjudgement.

The MD and PCA model can detect anomalies in the bearing system four days before failure, whereas the PARA-LSTM model can detect anomalies five days in advance. The PARA-LSTM model's ability to detect anomalies earlier is highly significant for timely spacecraft repair and maintenance.

Based on the previous analysis, it is evident that detecting anomalies greatly relies on selecting an appropriate threshold. The selection of this threshold is closely tied to choosing the model's hyperparameters, such as the learning rate, momentum factor, number of iterations,

and early stopping. This aspect also holds significant importance for future research endeavours.

Acknowledgements

This work was supported by a Key project of the Sichuan Science and Technology Department (2023ZHCG0020).

References

Aubert, S., Barnes, J.D. et al. (2022) 'Global Matrix 4.0 physical activity report card grades for children and adolescents: Results and analyses from 57 countries', *Journal of Physical Activity and Health*, Vol. 19, No. 11, pp.700–728.

Colombo, P., Dadalto, E. et al. (2022) 'Beyond Mahalanobis distance for textual OOD detection', *Advances in Neural Information Processing Systems*, Vol. 35, No. 35, pp.17744–17759.

Ding, X. and Feng, W. (2021) 'An anomaly detection method based on feature mining for wireless sensor networks', *International Journal of Sensor Networks*, Vol. 36, No. 3, pp.167–173.

Elhaik, E. (2022) 'Principal component analyses (PCA)-based findings in population genetic studies are highly biased and must be reevaluated', *Scientific Reports*, Vol. 12, No. 1, pp.14683.

Fisch, A.T., Eckley, I.A. et al. (2022) 'A linear time method for the detection of collective and point anomalies', *Statistical Analysis and Data Mining: The ASA Data Science Journal*, Vol. 15, No. 4, pp.494–508.

Gudovskiy, D., Ishizaka, S. et al. (2022) 'CFLOW-AD: real-time unsupervised anomaly detection with localization via conditional normalizing flows', *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp.98–107.

Han, S., Hu, X., Huang, H. et al. (2022) 'Adbench: anomaly detection benchmark', *Advances in Neural Information Processing Systems*, Vol. 35, No. 35, pp.32142–32159.

Haug, C.J. and Drazen, J.M. (2023) 'Artificial intelligence and machine learning in clinical medicine', *New England Journal of Medicine*, Vol. 388, No. 13, pp.1201–1208.

Kumar, K.E.A., Kalaga, D.V. et al. (2022) 'Comparative analysis of gated recurrent units (GRU), long short-term memory (LSTM) cells, autoregressive integrated moving average (ARIMA), seasonal autoregressive Integrated moving average (SARIMA) for forecasting COVID-19 trends', *Alexandria Engineering Journal*, Vol. 61, No. 10, pp.7585–7603.

Liu, C., Ma, Q. et al. (2022) 'A programmable diffractive deep neural network based on a digital-coding metasurface array', *Nature Electronics*, Vol. 5, No. 2, pp.113–122.

Lu, J., Ma, J., Zheng, X., Wang, G., Li, H. and Kiritsis, D. (2021) 'Design ontology supporting model-based systems engineering formalisms', *IEEE Systems Journal*, Vol. 16, No. 4, pp.5465–5476.

Nayak, D. and Perros, H. (2020) 'Automated real-time anomaly detection of temperature sensors through machine-learning', *International Journal of Sensor Networks*, Vol. 34, No. 3, pp.137–152.

Ozdemir, S.U.A.T. and Xiao, Y. (2013) 'FTDA: outlier detection-based fault-tolerant data aggregation for wireless sensor networks', *Security and Communication Networks*, Vol. 6, No. 6, pp.702–710.

Rumelhart, D. (1993)'Learning and connectionist representations', *Attention and Performance*, Vol. 14, No. 2, pp.3–30.

Santos, M.S., Abreu, P.H. et al. (2022) 'On the joint-effect of class imbalance and overlap: a critical review', *Artificial Intelligence Review*, Vol. 55, No. 8, pp.6207–6275.

Shayegan, M.J., Sabor, H.R. et al. (2022) 'A collective anomaly detection technique to detect crypto wallet frauds on bitcoin network', *Symmetry*, Vol. 14, No. 2, p.328.

Shen, Y., Wang, L. et al. (2023) 'Big-data and artificial-intelligence-assisted vault prediction and EVO-ICL size selection for myopia correction', *British Journal of Ophthalmology*, Vol. 107, No. 2, pp.201–206.

Sun, B., Osborne, L., Xiao, Y. and Guizani, S. (2007a) 'Intrusion detection techniques in mobile ad hoc and wireless sensor networks', *IEEE Wireless Communications*, Vol. 14, No. 5, pp.56–63.

Sun, B., Shan, X., Wu, K. and Xiao, Y. (2013) 'Anomaly detection based secure in-network aggregation for wireless sensor networks', *IEEE Systems Journal*, Vol. 1, No. 7, pp.13–25.

Sun, B., Wu, K., Xiao, Y. and Wang, R. (2007b) 'Integration of mobility and intrusion detection for wireless ad hoc networks', *International Journal of Communication Systems*, Vol. 20, No. 6, pp.695–721.

Sun, B., Yu, F., Wu, K., Xiao, Y. and Leung, V.C. (2006) 'Enhancing security using mobility-based anomaly detection in cellular mobile networks', *IEEE Transactions on Vehicular Technology*, Vol. 55, No. 4, pp.1385–1396.

Tanveer, M., Rajani, T. et al. (2022) 'Comprehensive review on twin support vector machines', *Annals of Operations Research*, Vol. 2, No. 4, pp.1–46.

Tokovarov, M. and Karczmarek, P. (2022) 'A probabilistic generalization of isolation forest', *Information Sciences*, Vol. 584, pp.433–449.

Uddin, S., Haque, I. et al. (2022) 'Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction', *Scientific Reports*, Vol. 12, No. 1, p.6256.

Wu, R., Deng, X., Lu, R. and Shen, X. (2015) 'Trust-based anomaly detection in emerging sensor networks', *International Journal of Distributed Sensor Networks*, Vol. 11, No. 10, p.363569.

Zhang, G., Li, Z. et al. (2022) 'efraudcom: an e-commerce fraud detection system via competitive graph neural networks', *ACM Transactions on Information Systems (TOIS)*, Vol. 40, No. 3, pp.1–29.

Zhao, H., Liu, H. et al. (2021) ' Feature extraction for data-driven remaining useful life prediction of rolling bearings', *IEEE Transactions on Instrumentation and Measurement*, Vol. 70, pp.1–10.

Zhou, J., Xiao, M., Niu, Y. and Ji, G. (2022) 'Rolling bearing fault diagnosis based on WGWOA-VMD-SVM', *Sensors*, Vol. 22, No. 16, p.6281.